

REMARKS

Claims 1-4, 13, 19, 34, 35, 46-49, 51, 52 and 59-61 are amended. Claims 1-4, 11-26, 28-42, 44-49, 51-57, 59-64, 66-72 and 74-76 are pending following entry of this amendment.

Summary of Examiner Interview

Applicant's representative and Examiner Tran held a telephonic interview on 1/14/2009. During the interview, Applicant's representatives and the Examiner discussed the rejections outstanding in the Examiner's office action and whether the cited references teach or suggest "operating system markers including a number of currently opened application programs" as claimed. No agreement was reached.

Claim Objections

Claims 1-4, 13, 19, 34, 35, 46-49, 51, 52 and 59-61 are objected to because of informalities. These claims have been amended as suggested by the Examiner. Accordingly, the Examiner is respectfully requested to withdraw this objection.

Claim 60 is further objected to for allegedly reciting "means for" components that are not clearly defined in the specification as being hardware (i.e., the Examiner interprets claim 60 as reciting software per se). Claim 60 recites a system for configuring a user interface of an application program and a user interface of an operating system of a computer system, the computer system including a plurality of application programs. Specifically, claim 60 recites, *inter alia*:

means for **storing** a plurality of application markers...
means for **storing** a plurality of operating system markers...; and
means executed by a **computer system**...

Claim 60 is supported throughout the specification, including, *e.g.*, at paragraph [0022]. Paragraph [0022] recites, *inter alia*:

Referring now to Fig. 1, there is shown a block diagram depicting a system for practicing the present invention according to one embodiment. The various functional components shown therein can be implemented, for example, in a conventional **personal computer** such as a Power Macintosh G4 running an operating system (OS) 101 such as MacOS X (both available from Apple Computer, Inc., of Cupertino, California), along with one or more applications 102. For illustrative purposes, three applications 102 are shown in Fig. 1, although one skilled in the art will recognize that any number of applications 102 may be running on the **computer system** at any given time. (emphasis added)

Thus, paragraph [0022] of the specification makes clear that claim 60 recites “means for” components executed by a computer system. Components executed by a computer system are well established as statutory subject matter. *See* MPEP 2106. Accordingly, the Examiner is respectfully requested to withdraw this objection.

During the interview of 1/14/2009 the Examiner requested that independent claims 1, 46, 60 and 61 be amended to recite “operating markers including a numerical number of how many application programs are currently opened” instead of “operating markers including **a number** of currently opened applications programs” (emphasis added) as currently recited. The Examiner stated that the rationale for this request is that it is unclear whether “a number” as currently recited means “a count” or “a number.” The Applicant respectfully disagrees with this rationale. An ordinary English word whose meaning is clear and unquestionable must be construed by Examiner to mean exactly what it says. *See* MPEP 2111.01. Here, the meaning of “a number” is clear and unquestionable. Thus, “a number” means “a number.”

35 U.S.C. § 103 Rejections

Claims 1-4, 13-23, 30, 34-40, 42, 44-49, 51-57, 59-64, 66-72, and 74 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Lanier (No. RE37,431E) in view of Hoffberg (6,400,996). Claims 26, 28, 29, 33, 75 and 76 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hoffberg in view of Morrison (No. 2003/0030668). Claims 24, 31 and 32 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Hoffberg in view of Aleksander (No. 7,080,321). Claim 41 stands rejected under under 35 U.S.C. § 103(a) as unpatentable over Hoffberg in view of Lehmeier (No. 6,981,242). These rejections are now traversed.

The independent claims recite elements related to configuring user interfaces in a computer system responsive to a user proficiency level. One indication of a user's proficiency level is the number of application programs the user has open at a given time. Less proficient computer users will tend to have a smaller number of application programs running, whereas more proficient computer users will tend to have a larger number of applications running. Thus, independent claims 1 and 75 recite, *inter alia*, determining a user proficiency level with respect to the user interface of an operating system based on **a number of currently opened application programs**. Independent claims 46, 60, 61 and 76 recite similar elements. For example, independent claim 1 recites:

- storing a plurality of application program markers, each application program marker associated with one of the plurality of application programs, and indicating a user interaction with the associated one of the application programs;
- storing a plurality of operating system markers, each operating system marker indicating a user interaction with the operating system, the operating system markers including an indication of **a number of currently opened application programs**;
- assigning weights to each of the plurality of application program markers and each of the plurality of operating system markers;

- determining a weighted score as a function of a subset of the weighted operating system markers and a subset of the weighted application program markers;
- determining a user proficiency level with respect to the user interface of the application program and the user interface of the operating system based upon the weighted score; and
- automatically configuring at least one functional component of the user interface of the application program and at least one functional component of the user interface of the operating system responsive to the user proficiency level.

Similarly, claim 75 recites:

- determining a user proficiency level with respect to the user interface of the application program and user interface of the operating system based upon **a number of application programs currently open**, a historical average number of concurrently open applications, a number of windows currently open, and a historical average number of concurrently open windows; and
- automatically configuring at least one functional component of the user interface of the application program and at least one functional component of the user interface of the operating system responsive to the user proficiency level.

Independent claims 46, 60, 61 and 76 recite similar elements. The cited references do not disclose or suggest determining a user proficiency level with respect to the user interface of an operating system based on a number of currently opened application programs.

Lanier describes a help system for providing help text to a computer user. *See* Lanier, Abstract and Figure 3. The help text provides users with information about how to perform tasks in an application. The user accesses the help text by clicking items in the application's menubar. As the user performs tasks in the application the help system creates a record of each task completed by the user. If a completed task corresponds to a task for which help text has been defined, then the help system concludes that the user no longer needs help performing that task. The help system therefore modifies the menubar to exclude items for accessing help text for the completed task. *See* Lanier, column 3, lines 13-20, column 4, lines 7-17 and column 5, lines 18-35. The help system also

modifies the menubar each time the user accesses a different part of the application so that the available help text is relevant to the part of the application being accessed by the user. *See* Lanier, column 1, lines 39-43.

The Examiner acknowledges that Lanier does not teach or suggest configuring at least one functional component of the user interface of the application program and at least one functional component of the user interface of the operating system responsive to a user proficiency level. *See* Office Action dated 11/26/2006 (“OA”), page 5. In addition, Lanier also does not disclose or suggest determining a user proficiency level with respect to the user interface of an operating system based on a number of currently opened application programs.

The Examiner indicates that Lanier discloses determining a user proficiency level with respect to the user interface of an operating system based on a number of currently opened application programs at column 5, lines 37-49 and column 6, lines 1-9. These portions of the reference provide additional details about how Lanier’s help system modifies the menubar each time the user accesses a different part of the application so that the available help text is relevant to the part of the application being accessed by the user. Specifically, column 5, lines 19-37 discloses an event interpreter for monitoring user activity in the operating environment, and that the operating environment is divided into various levels in order to assist the event interpreter in determining which portion of the application the user is accessing. Column 5, lines 37-49 describes the information required by the event interpreter in order to determine which part of the application the user is accessing. This information includes whether an application is running or has quit, and the identity of a running application. However, the event interpreter does not require information about the number of applications that are currently running. Column

6, lines 1-9 discloses that the event interpreter can make a distinction between whether the user is accessing an application or an accessory for the application. Thus, at most Lanier discloses an event interpreter that provides information to the help system for modifying the menubar based on the different parts of the application program the user is accessing. However, Lanier does not disclose or suggest determining a user proficiency level with respect to the user interface of an operating system based on the number of currently opened application programs.

Morrison does not remedy the deficiencies of Lanier. Morrison describes a help system having a set of help files. The help system keeps a history log of which help files have been accessed by the user. Once a user has accessed a help file, the help system determines that the user no longer needs to access that file again. Thus, the help system customizes the presentation of the help files to exclude previously accessed files. *See* Morrison, Abstract and paragraph [0010]. However, Morrison does not disclose or suggest determining a user proficiency level with respect to the user interface of an operating system based on the number of currently opened application programs.

The Examiner cites Morrison at paragraphs [0030] and [0028] as disclosing determining a user proficiency level with respect to the user interface of an operating system based on a number of currently opened application programs. This is incorrect. Here, Morrison merely provides further detail about how the help system keeps a history log of which help files have been accessed by the user. Paragraphs [0027] and [0028] describe script code that is embedded in each help file. When a user opens a help file, the script code generates a cookie. Each cookie has an ID to indicate that the user has opened a particular help file. The cookies are combined with timestamps to generate a history log of every help file that has been opened by the user. Paragraph [0030]

discloses that the help system can make inferences about the skill level of the user in a given topic after the user has opened several help files related to that topic. The Examiner indicates that these skill level inferences are based on the number of help files that the user has opened, and thus Morrison discloses determining a user proficiency level with respect to the user interface of an operating system based on a number of currently opened application programs. *See* OA, page 13.

However, Morrison does not disclose that these skill level inferences are based on a number of help files the user has currently opened. Instead, Morrison merely discloses that skill level cannot be inferred until a certain number of files have been opened. Thus, although the number of opened files may trigger the inference, the inference itself is not based on the number of files. Furthermore, Morrison appears to indicate that the skill level inferences have nothing to do with a number of currently opened help files. Specifically, Morrison discloses that the skill level inferences are “based on an experience level of a user indicated by the help history file or a type of material within the help source files” (emphasis added). *See* Morrison, paragraph [0052], last sentence, and paragraph [0029]. Thus, at most Morrison discloses inferring a user’s skill level for a particular topic based on the type of material contained within previously opened help files. Moreover, even if Morrison did infer a user’s skill level based on a number of currently open files, this still does not disclose or suggest determining a user proficiency level based on a number of currently open applications. Thus, Morrison does not disclose or suggest determining a user proficiency level with respect to the user interface of an operating system based on **the number of currently opened application programs**.

Hoffberg, Lehmeier, Abbott and Aleksander also do not teach or suggest determining a user proficiency level with respect to the user interface of an operating

system based on **a number of currently opened application programs**. The combination of Lanier with any of these references therefore does not teach or suggest all elements of the independent claims 1, 46, 60, 61, 75 and 76 nor the claims that depend therefrom.

Conclusion

Should the Examiner wish to discuss the above amendments and remarks, or if the Examiner believes that for any reason direct contact with Applicant's representative would help to advance the prosecution of this case to finality, the Examiner is invited to telephone the undersigned at the number given below.

Respectfully submitted,
JESSICA KAHN

Dated: March 9, 2009

/Robert R. Sachs/

Robert R. Sachs, Reg. No. 42,120
Attorney for Applicant
Fenwick & West LLP
Silicon Valley Center
801 California Street
Mountain View, CA 94041
Tel.: (415) 875-2410
Fax: (650) 938-5200